Problem Set 4

This problem set is due on *Thursday, October 11, 2001* at the beginning of class. Late homeworks will *not* be accepted. You are to work on this problem set in groups of three or four people.

Mark the top of each sheet with your names, 6.857, the problem set number and question, and the date. **Type up your solutions** and be clear. Each problem should begin on a new sheet of paper. That is, you will turn in each problem on a separate pile of paper.

Problem 4-1. Vote all you want

After reading the FOO scheme, Alyssa P. Hacker decided to create a new version called the FOO-BAR scheme. Alyssa wants to augment the properties of FOO to allow voters to vote as many times as they like, with only their last vote actually counting. Help Alyssa make the modifications to the original FOO scheme to support this new feature. Make sure to argue how your new scheme still maintains Completeness, Soundness, Privacy, Eligibility, Fairness, and Verifiability. In terms of Unreusability, argue how although a voter can vote several times, only their last vote is counted.

Hint: There are several approaches to this problem. In one scheme that we envision, the voter asks the Authority sign a newly generated public key instead of a commitment to a ballot.

Problem 4-2. Weak Randomness and ElGamal

Recall that the ElGamal signature scheme is randomized. Obviously the key generation involves randomization. But unlike the signature generation process in RSA, the ElGamal signature generation process relies on a source of good randomness too. When this source of randomness is predictable, the security of ElGamal can break down.

We repeat for your convenience the ElGamal signature scheme.

Key generation:

1. Generate a large **random** prime p and a generator α of the multiplicative group Z_p^*
2. Select a random integer a such that $1 \leq a \leq p-2$.
3. Compute $y = \alpha^a \pmod{p}$
4. The public key is $(p,\ \alpha,\ y)$ and the private key is a.

To sign a message m:

1. Select a *random secret* integer k such that $1 \leq k \leq p-2$ and $gcd(k,\ p-1) = 1$.
2. Compute $r = \alpha^k \pmod{p}$.
3. Compute $k^{-1} \pmod{p-1}$.
4. Compute $s = k^{-1}\{h(m) - ar\} \pmod{p-1}$.
5. Output the pair $(r,\ s)$ as the signature of m.

To verify a signature $(r,\ s)$ on m:

1. Obtain the public key $(p,\ \alpha,\ y)$.
2. Verify that $1 \le r \le p-1$. Otherwise reject the signature.
3. Compute $v_1 = y^r r^s \pmod{p}$.
4. Compute $h(m)$ and $v_2 = \alpha^{h(m)} \pmod{p}$.
5. Accept the signature iff $v_1 = v_2$.

See section 11.5.2 of Handout 7 or lecture notes 6 for discussion of the ElGamal signature scheme. In this problem set, we use the notation from handout 7.

(a) Reusing k

Explain how to recover the secret key a when the signer reuses the same k for two different messages. Write an equation for a. That is, extract the private key from the public key and two (message, signature) pairs. State your assumptions.

(b) Generating k with a linear congruential number generator

Louis Reasoner was unable to afford the advanced Lava Lamp random number generator – nor would it fit in his wallet-sized smartcard. Instead, he seeds a linear congruential number generator[1] with a truly random number, k_0. During the signature of the ith message in Louis' scheme, $k =$

1. If $i = 0$, return a truly random number k_0 where $3 \le k_0 \le p-2$ and $gcd(k_0,\ p-1) = 1$.
2. Otherwise let $k = k_{i-1}$.
3. Do $k \leftarrow k + z \pmod{p-1}$ until $gcd(k,\ p-1) = 1$.
4. Remember $k_i = k$ and return k.

z is a secret, truly random even number that is the same for every signature generation. This LCNG remembers its output for its next execution. Assume that p is a safe Sophie-Germain prime in that $p = 2q + 1$ where q is prime.

Explain how an adversary can break the system after seeing many sequential (message, signature) pairs. Write an equation for a. Note that your attack need not work 100% of the time, but it should work most of the time assuming that certain values are invertible $\pmod{p-1}$. State your assumptions about which values must be invertible and any other conditions for your attack to work. Discuss how many (message, signature) pairs you will need to break the scheme.

Problem 4-3. Holiday

In honor of the 4-day weekend, there is no problem 3. Instead, we offer you a couple marginal jokes. For your protection, we have triple encrypted the answers with a special cipher. :-)

Why do cryptographers like to watch TV at night?
Orpnhfr bgurejvfr gurl jbhyq zvff gur cevzr gvzr arjf.

What do you call it when an automaton sounds like it computes?
Nhgbzngncbrvn.

[1]Thousands of Web sites use LCNGs to generate cookie-based Session IDs on the Web. Is this a good idea?